Отчёт по лабораторной работе №3 по дисциплине Компьютерный практикум по статистическому анализу данных

Управляющие структуры

Шаповалова Диана Дмитриевна

Содержание

# 1 Цель работы

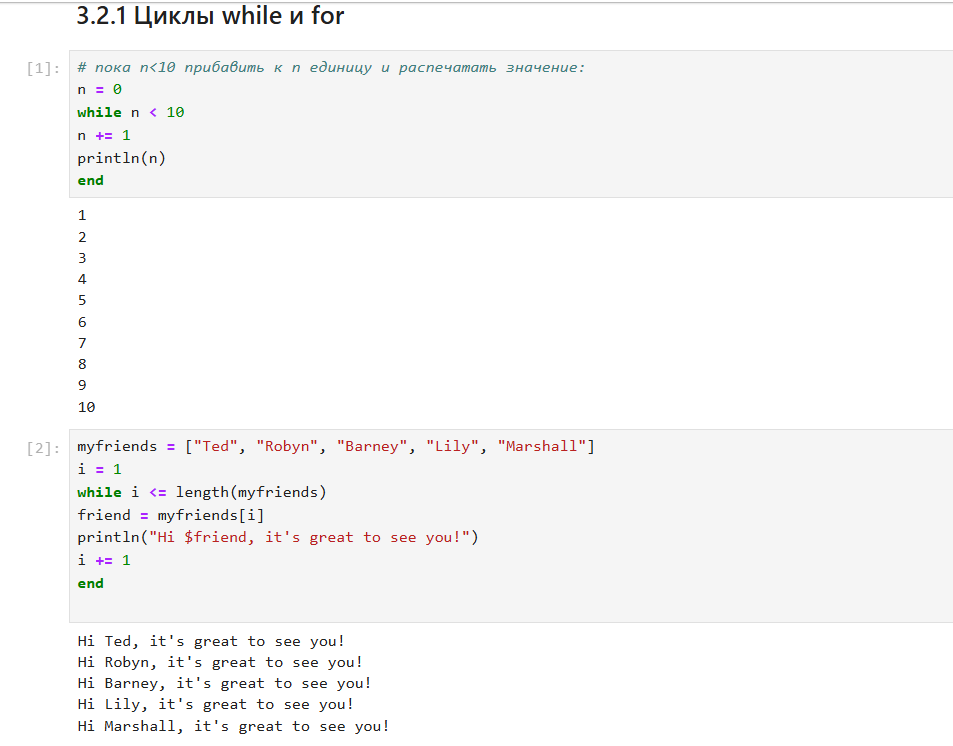
Основная цель работы — освоить применение циклов функций и сторонних для Julia пакетов для решения задач линейной алгебры и работы с матрицами.

# 2 Выполнение лабораторной работы

## 2.1 Циклы while и for

Для различных операций, связанных с перебором индексируемых элементов структур данных, традиционно используются циклы while и for. Например, while можно использовать для формирования элементов массива:

Другой пример демонстрирует использование while при работе со строковыми элементами массива, подставляя имя из массива в заданную строку приветствия и выводя получившуюся конструкцию на экран

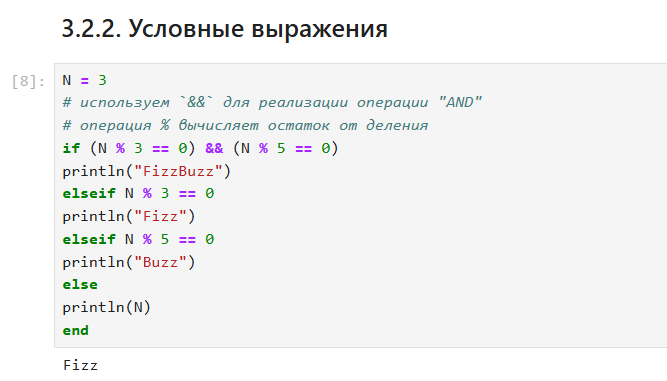


Выполняем примеры по циклам while и for

## 2.2 Условные выражения

Довольно часто при решении задач требуется проверить выполнение тех или иных условий. Для этого используют условные выражения.

Например, пусть для заданного числа 𝑁 требуется вывести слово «Fizz», если 𝑁 делится на 3, «Buzz», если 𝑁 делится на 5, и «FizzBuzz», если 𝑁 делится на 3 и 5



Выполняем примеры по условным выражениям

## 2.3 Функции

Julia дает нам несколько разных способов написать функцию. Первый требует ключевых слов function и end.

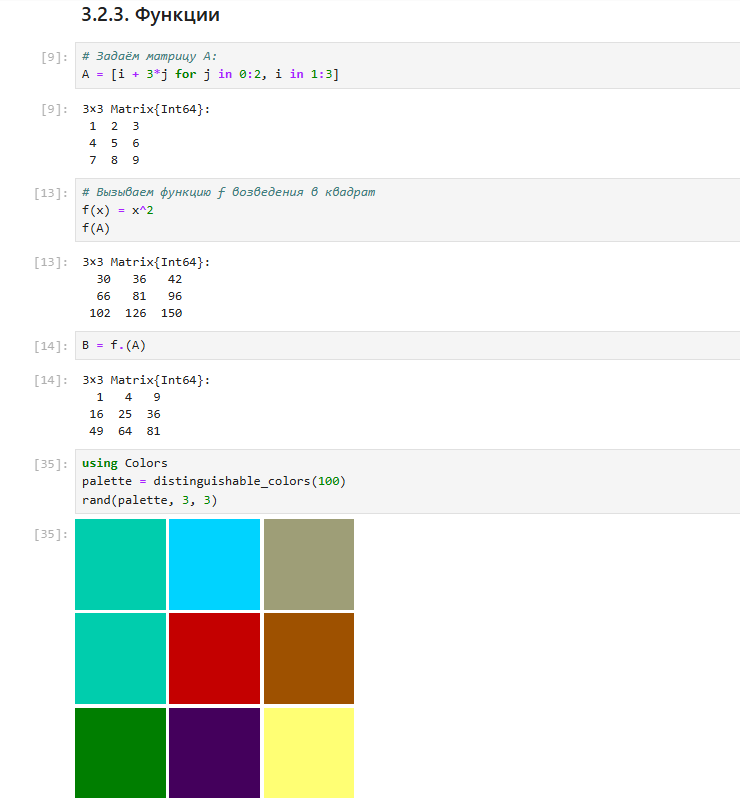
В Julia функция map является функцией высшего порядка, которая принимает функцию в качестве одного из своих входных аргументов и применяет эту функцию к каждому элементу структуры данных, которая ей передаётся также в качестве аргумента.

Функция broadcast — ещё одна функция высшего порядка в Julia, представляющая собой обобщение функции map.Функция broadcast() будет пытаться привести все объекты к общему измерению, map() будет напрямую применять данную функцию поэлементно. Синтаксис для вызова broadcast такой же, как и для вызова map.

## 2.4 Сторонние библиотеки (пакеты) в Julia

Julia имеет более 2000 зарегистрированных пакетов, что делает их огромной частью экосистемы Julia. Есть вызовы функций первого класса для других языков, обеспечивающие интерфейсы сторонних функций. Можно вызвать функции из Python или R, например, с помощью PyCall или Rcall.

Например, добавим и загрузим пакет Colors, затем создадим палитру из 100 разных цветов, а затем определим матрицу 3 × 3 с элементами в форме случайного цвета из палитры, используя функцию rand.

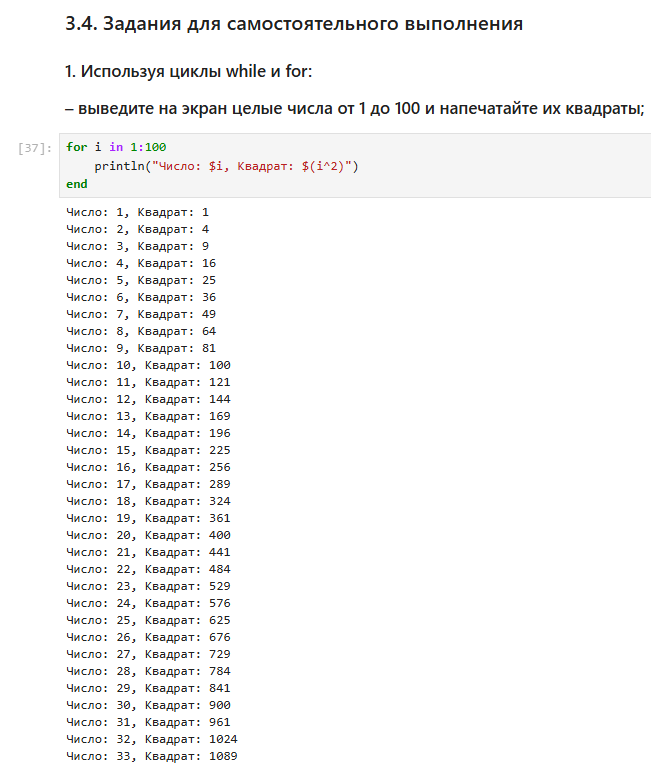


Выполняем примеры по функциям и сторонним библиотекам

## 2.5 Задания для самостоятельного выполнения

1. Используя циклы while и for:

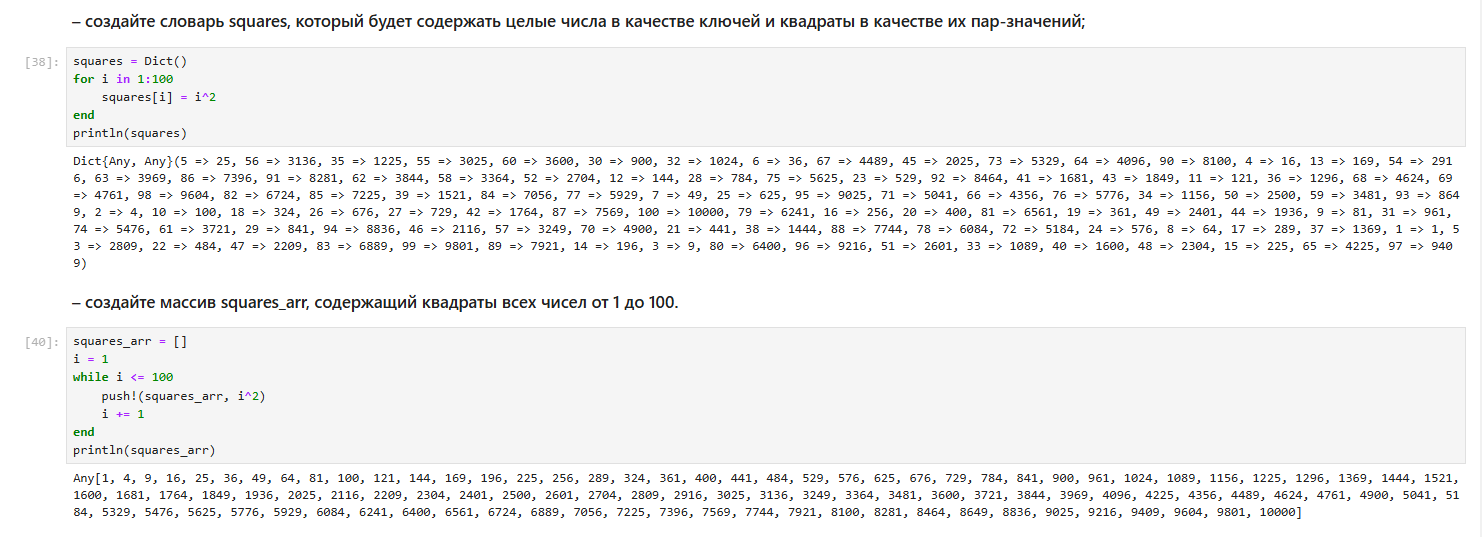
– выведите на экран целые числа от 1 до 100 и напечатайте их квадраты;



Задание №1

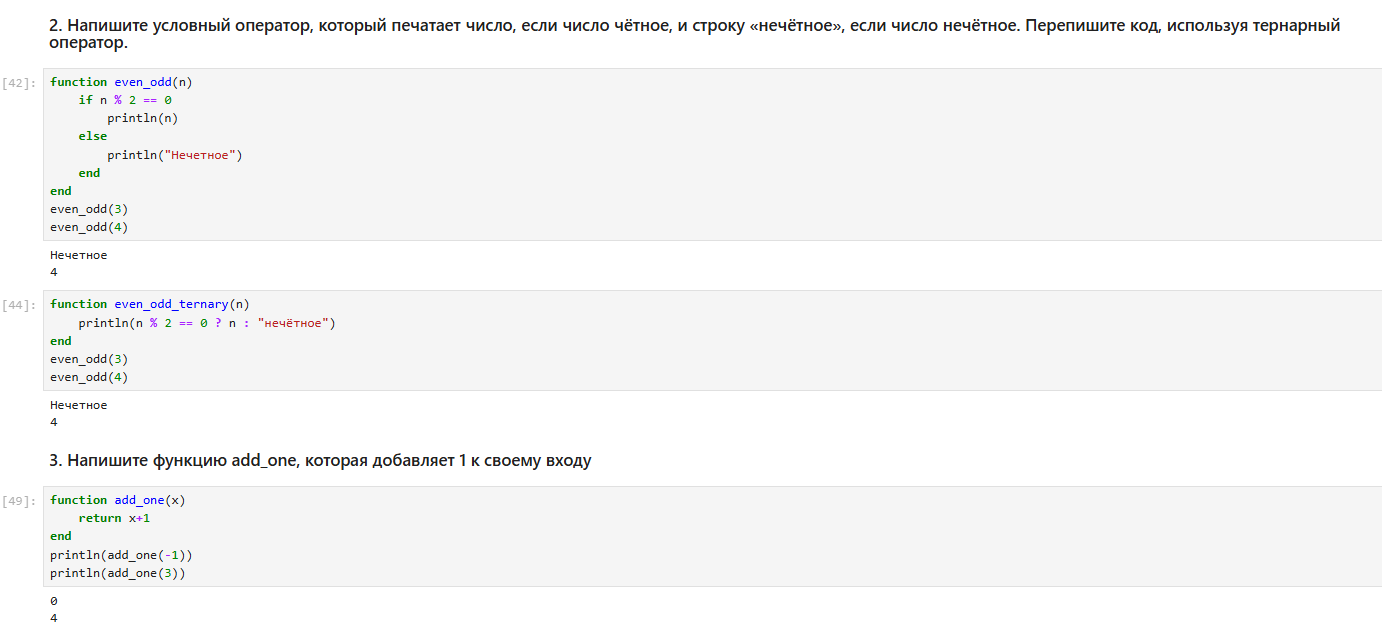
– создайте словарь squares, который будет содержать целые числа в качестве ключей и квадраты в качестве их пар-значений;

– создайте массив squares\_arr, содержащий квадраты всех чисел от 1 до 100.



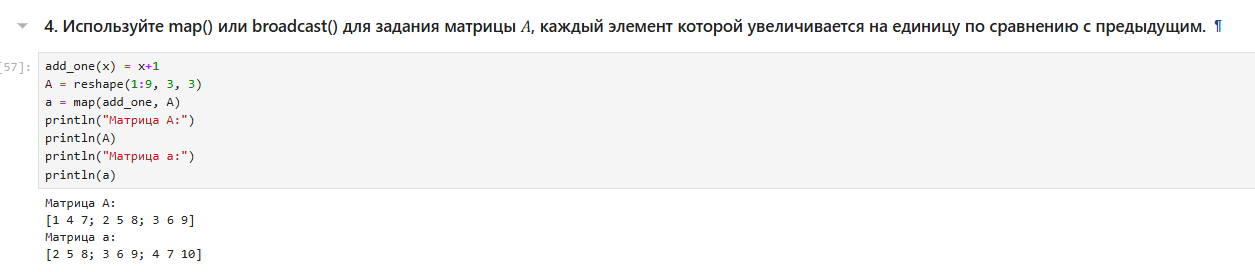
Задание №1

1. Напишите условный оператор, который печатает число, если число чётное, и строку «нечётное», если число нечётное. Перепишите код, используя тернарный оператор.
2. Напишите функцию add\_one, которая добавляет 1 к своему входу.



Задание №2 и 3

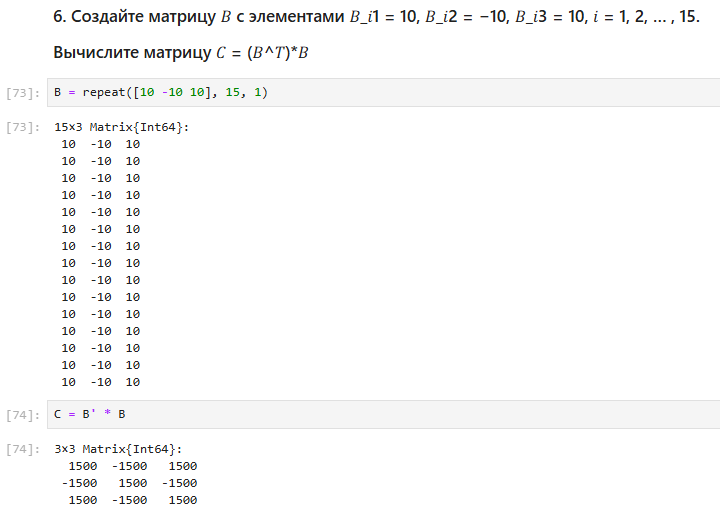
1. Используйте map() или broadcast() для задания матрицы 𝐴, каждый элемент которой увеличивается на единицу по сравнению с предыдущим.



Задание №4

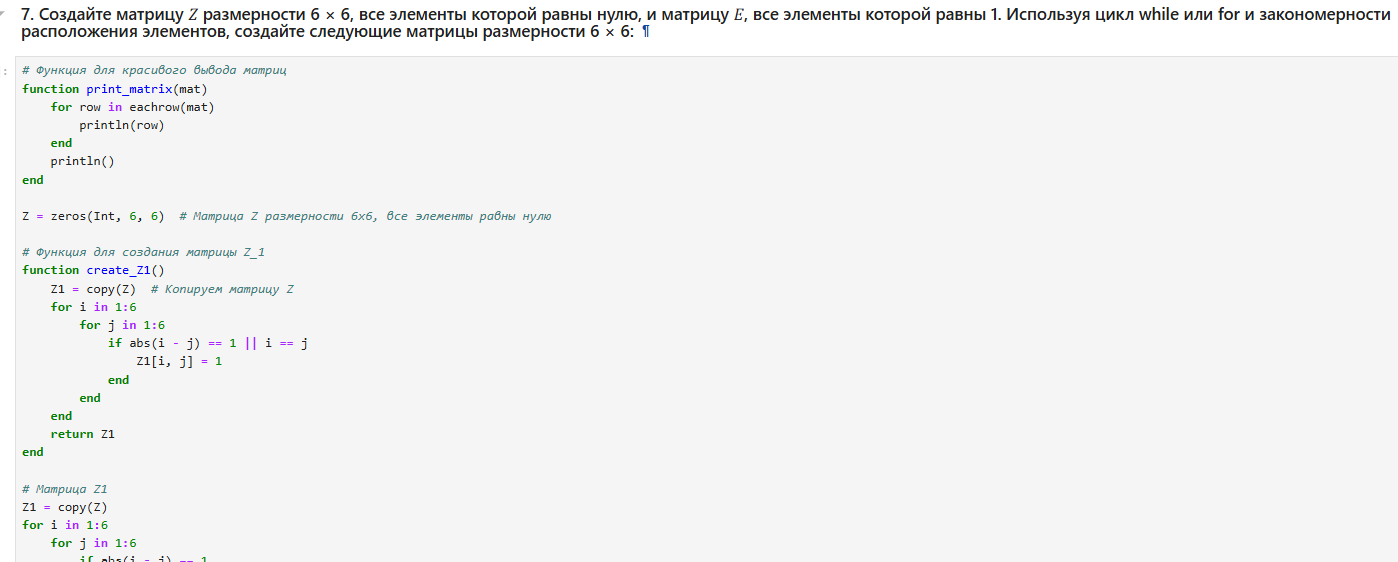
1. Создайте матрицу 𝐵 с элементами 𝐵𝑖1 = 10, 𝐵𝑖2 = −10, 𝐵𝑖3 = 10, 𝑖 = 1, 2, … , 15.

Вычислите матрицу 𝐶 = 𝐵 \* 𝐵^𝑇



Задание №6

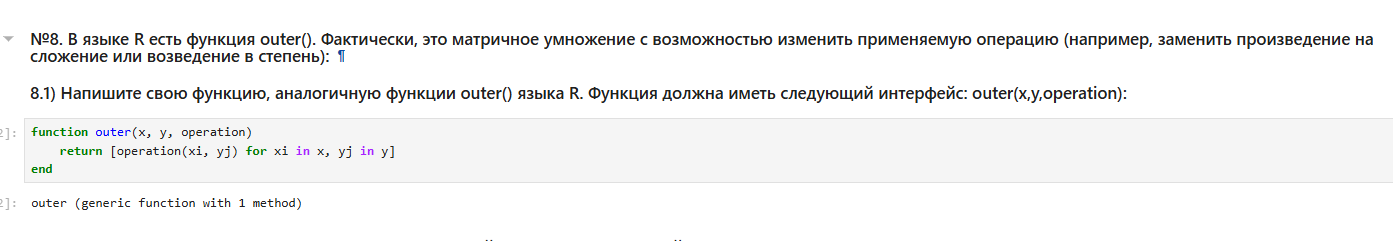
1. Создайте матрицу 𝑍 размерности 6 × 6, все элементы которой равны нулю, и матрицу 𝐸, все элементы которой равны 1. Используя цикл while или for и закономерности расположения элементов, создайте следующие матрицы размерности 6 × 6



Задание №7

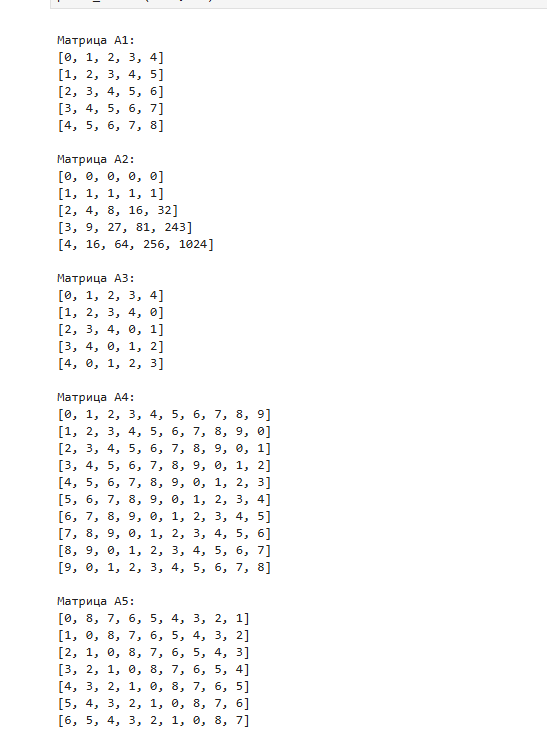
1. В языке R есть функция outer(). Фактически, это матричное умножение с возможностью изменить применяемую операцию (например, заменить произведение на сложение или возведение в степень).

– Напишите свою функцию, аналогичную функции outer() языка R. Функция должна иметь следующий интерфейс: outer(x,y,operation).



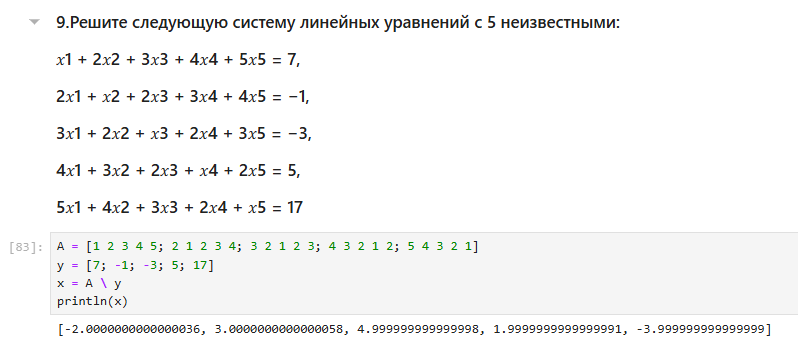
Задание №8

– Используя написанную вами функцию outer(), создайте матрицы следующей структуры



Задание №8

1. Решите следующую систему линейных уравнений с 5 неизвестными

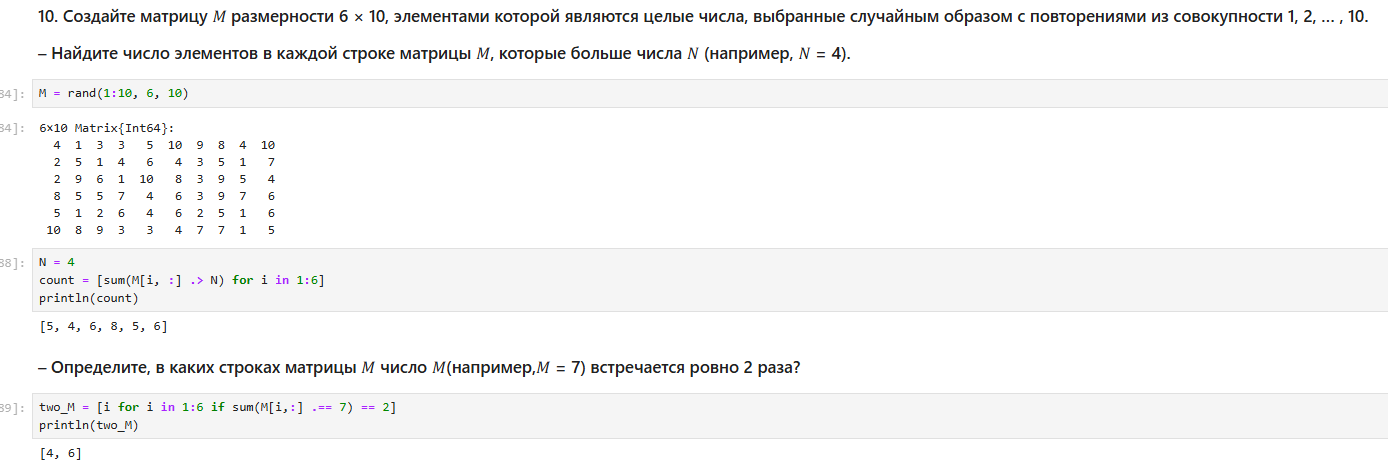


Задание №9

1. Создайте матрицу 𝑀 размерности 6 × 10, элементами которой являются целые числа, выбранные случайным образом с повторениями из совокупности 1, 2, … , 10.

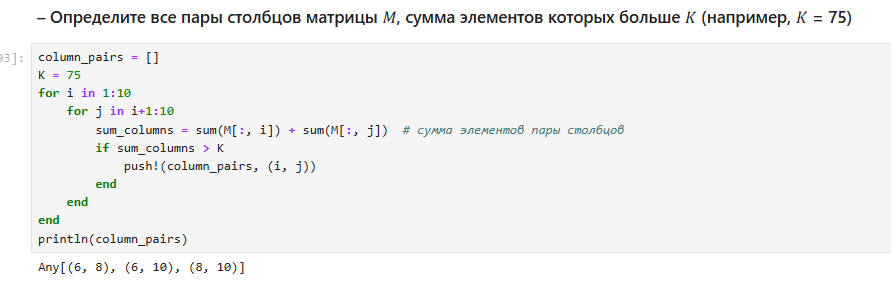
– Найдите число элементов в каждой строке матрицы 𝑀, которые больше числа 𝑁 (например, 𝑁 = 4).

– Определите, в каких строках матрицы𝑀число𝑀(например,𝑀 = 7) встречается ровно 2 раза?



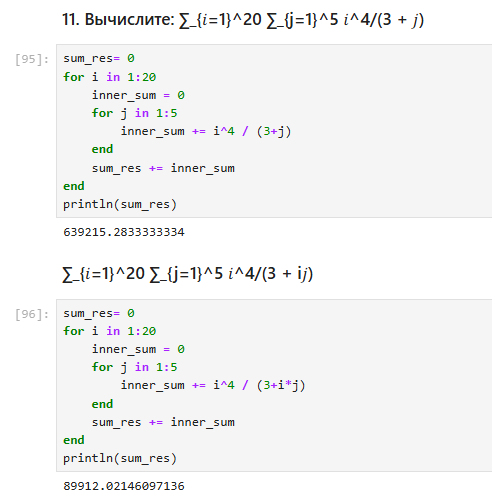
Задание №10

– Определите все пары столбцов матрицы 𝑀, сумма элементов которых больше 𝐾 (например, 𝐾 = 75).



Задание №10

1. Вычислите



Задание №11

# 3 Выводы

Мы освоили применение циклов функций и сторонних для Julia пакетов для решения задач линейной алгебры и работы с матрицами.